



Mi vagyunk a Te jövőd!

Programfejlesztés

C#.NET és MySQL alapú adatkezelő alkalmazás

A projekt keretében egy C#.NET alapú adatkezelő alkalmazást készítettem, amely MySQL adatbázissal kommunikál. A célom az volt, hogy egy egyszerű felhasználói felületen keresztül lehessen adatokat felvinni, módosítani, törölni és megjeleníteni az adatbázisból.

A fejlesztést Visual Studio környezetben végeztem, és a MySQL adatbázishoz a MySql.Data nevű könyvtár segítségével csatlakoztam. Az alkalmazásban létrehoztam űrlapokat, ahol különböző adatokat lehetett megadni, majd gombok segítségével lehetett ezeket az adatbázisba menteni vagy frissíteni. A táblázatos megjelenítést DataGridView komponenssel oldottam meg, így az adatok átláthatóan listázhatók voltak.

Adatbázis létrehozása

Az alkalmazás fejlesztése előtt első lépésként létre kellett hoznom az adatbázist, amelyet a program később használni fog. Ehhez a MySQL Workbench-et használtam, ahol létrehoztam egy új adatbázist a megfelelő névvel.

Az adatbázisban egy vagy több táblát is létrehoztam, attól függően, hogy milyen adatokat kellett tárolni. A táblák mezőit úgy terveztem meg, hogy minden fontos adat elérjen bennük, például

szöveges adatokhoz VARCHAR típusú mezőket, számokhoz pedig INT típusú mezőket használtam. Beállítottam az elsődleges kulcsokat (PRIMARY KEY), hogy minden rekordot egyértelműen azonosítani lehessen.

A konfigurálás során ügyeltem arra, hogy a mezők megfelelő hosszal legyenek létrehozva, valamint hogy kötelező (NOT NULL) mezőket is beállítsak, ahol az adatok nem maradhatnak üresen. Emellett még létrehoztam szükség szerint idegen kulcsokat (FOREIGN KEY) is, ha több táblát kellett összekapcsolni.



Forrás: Saját forrás

rollno (egész szám, elsődleges kulcs, automatikusan növekvő)

name (szöveg típusú, a személy neve)

address (szöveg típusú, a személy címe)

phone (szöveg vagy szám típusú, telefonszám)

email (szöveg típusú, e-mail cím)

C# Windows Form alkalmazás létrehozása

A Visual Studio-ban egy új Windows Forms alkalmazást hozunk létre C# nyelven. Az űrlapon elhelyezünk címkéket (Label), szövegmezőket (TextBox) a felhasználói adatok beviteléhez, valamint gombokat (Button) az adatok beszúrásához, frissítéséhez, törléséhez és megtekintéséhez. Emellett egy DataGridView vezérlőt is hozzáadunk, amely az adatbázisban tárolt adatokat jeleníti meg vizuálisan.

Rollno	Name	Address	Phone	Email
*				

Forrás: Saját Forrás

Adatok megjelenítése

Miután sikeresen létrehoztam az adatbázist és beállítottam a kapcsolatot, a következő lépés az adatok megjelenítése és kezelése volt az alkalmazásban. Ehhez a C#.NET környezetben egy DataGridView komponenst használtam, ami lehetővé tette, hogy az adatokat táblázatos formában jelenítsem meg a felhasználónak.

A megjelenítéshez SQL SELECT lekérdezést használtam, amit a MySQL adatbázisba küldtem, majd az eredményeket egy DataTable segítségével töltöttem be a DataGridView-ba. Így a program elindításakor azonnal láthatók voltak a meglévő adatok.

Önreflexió: A projekt során sok új dolgot tanultam meg a programfejlesztés és az adatbáziskezelés területén. Korábban még nem készítettem olyan alkalmazást, amely valós adatbázissal kommunikált volna, ezért eleinte kihívásnak éreztem a MySQL és a C# közötti kapcsolat beállítását. Viszont ahogy egyre többet gyakoroltam, egyre jobban megértettem az adatbázisok működését és az SQL lekérdezések helyes használatát. Különösen hasznos volt számomra az adatkezelési műveletek (új rekord hozzáadása, módosítás, törlés) megvalósítása, mert így gyakorlatban is láthattam, hogyan zajlik egy CRUD (Create, Read, Update, Delete) rendszer működése. A DataGridView komponenssel való munka is segített abban, hogy átláthatóbbá és felhasználóbarátabbá tegyem az alkalmazásomat.

